(19)日本国特許庁（ＪＰ）　(12)公開特許公報（A）　(11)特許出願公開番号

特開平11－317735

(43)公開日　平成11年(1999)11月16日

(51)Int.Cl.[6]	識別記号	FI	
H04L 9/08		H04L 9/00	601F
G06F 13/00	351	G06F 13/00	351Z
G09C 1/00	640	G09C 1/00	640D
H04L 9/32		H04L 9/00	673E
			673B

審査請求　未請求　請求項の数25　ＯＬ　（全 17 頁）　最終頁に続く

(21)出願番号　特願平11－36453

(22)出願日　平成11年(1999)２月15日

(31)優先権主張番号　０２４９２８

(32)優先日　1998年２月17日

(33)優先権主張国　米国（ＵＳ）

(71)出願人　599021000
シーサラマン　ラマスブラマニ
SEETHARAMAN RAMASUBRAMANI
アメリカ合衆国，カリフォルニア州
95129　サン・ホゼ　ミラー・アヴェニュ　1195

(72)発明者　シーサラマン　ラマスブラマニ
アメリカ合衆国，カリフォルニア州
95129　サン・ホゼ　ミラー・アヴェニュ　1195

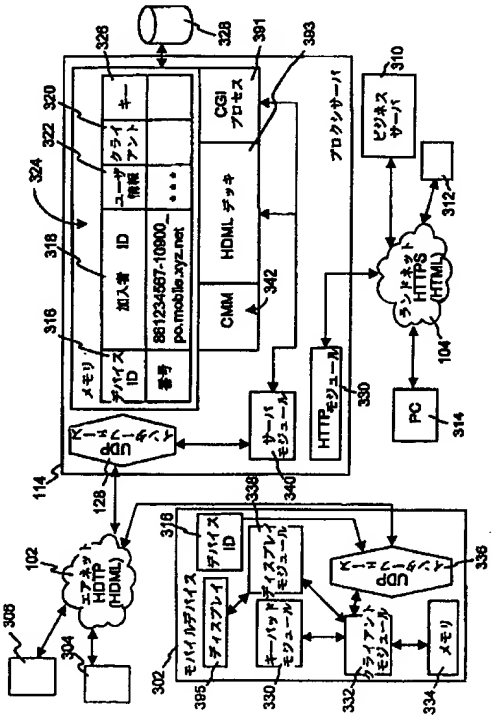
(74)代理人　弁理士　伊東　忠彦　（外１名）

(54)【発明の名称】　データネットワークにおける２方向インターラクティブコミュニケーションデバイスのための集中証明書管理システム

(57)【要約】

【課題】　データネットワークにおける貧弱なクライアントデバイスのための集中証明書管理システムを提供する。

【解決手段】　本発明は、サーバデバイスに貧弱なクライアントデバイスのためのデジタル証明書を管理させる証明書管理モジュールを有する。各貧弱なクライアントデバイスのために証明書を取得する遅延を最小にするために、その証明書管理モジュールは認証局により署名された固定数の自由証明書と各秘密キーを証明書データベースの中に保持し、証明書更新メッセージに応じて頻繁に自由証明書を更新する。貧弱なクライアントデバイスのためにユーザアカウントが生成される度に、証明書管理モジュールは１つ又はそれ以上の自由証明書を証明書データベースから取得し、取得した証明書を生成されたアカウントに関連付け、その間、証明書管理モジュールは、証明書データベースを埋めるために認証局とともに新しい自由証明書を作成する。



【特許請求の範囲】

【請求項１】　データネットワークを介してプロクシサーバと接続された複数の貧弱（thin）なクライアントデバイスのための、該プロクシサーバデバイスにおける集中証明書を管理するための方法であって、

該方法は前記プロクシサーバによりアクセス可能な自由証明書データベースを保持し、該自由証明書データベースは認証局（ＣＡ）によって発行された複数の自由証明書を有し、各自由証明書は対応する公開キー及び対応する秘密キーを有し、

該方法は前記プロクシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記各貧弱なクライアントデバイスは複数のユーザアカウントのうちの一つと関連付けられ、各ユーザアカウントは該ユーザアカウントに割り当てられたデバイスＩＤ、公開及び秘密キーのリスト、及び、該ユーザアカウントに割り当てられた証明書のリストを有し、

該方法は前記証明書データベースから取り出された少なくとも一つの証明書を前記ユーザアカウントデータベースにおける各ユーザアカウントに加えることを特徴とする方法。

【請求項２】　前記プロクシサーバにおける証明書データベースの保持は、

前記証明書データベースにおける自由証明書の数が下の閾値より低い場合に証明書要求を受け、

新たな証明書を生成し、該新たな証明書の生成は、

該新たな証明書のための識別名を生成し、

該新たな証明書のための新たな公開キー及び新たな秘密キーを生成し、

前記証明書要求を前記ＣＡに送信し、該証明書要求は生成された該新たな公開キーを有し、

ＣＡにより署名された新たな証明書を受信し、

前記自由証明書データベースに前記新たな証明書を置くことからなる請求項１に記載の方法。

【請求項３】　前記ユーザアカウントデータベースの保持は、

新たな貧弱なクライアントデバイスが活性化されると、前記自由証明書データベースから前記自由証明書のうちの一つを取得し、

新たなデバイスＩＤ及び新たな加入者ＩＤに関連した新たなユーザアカウントを構築し、

取得した前記自由証明書、対応する秘密キー及び公開キーを前記新たなユーザアカウントと関連付けることからなる請求項１に記載の方法。

【請求項４】　証明書更新要求を受信すると前記自由証明書データベースにおける前記自由証明書を更新することを更に有する請求項１に記載の方法。

【請求項５】　前記証明書更新要求を受信すると前記自由証明書データベースにおける自由証明書を更新することは、

該証明書更新要求が証明書取り消しリストである場合に、前記自由証明書データベースから無効な証明書を取り除くことからなる請求項４に記載の方法。

【請求項６】　前記証明書更新要求を受信すると自由証明書データベースにおける自由証明書を更新することは、

前記証明書更新要求における挿入／削除クエリーに応じて前記自由証明書データベースから証明書を削除することからなる請求項４に記載の方法。

【請求項７】　有効なデバイスＩＤを有する貧弱なクライアントから新たに設定されたユーザ名とパスワードを受信すると、該有効なデバイスＩＤと関連付けられた前記ユーザアカウントデータベースにおけるユーザアカウントを更新することを含む請求項１に記載の方法。

【請求項８】　前記ユーザアカウントデータベースにおける前記ユーザアカウントは、インターネットを介して前記プロクシサーバに接続されたコンピュータから前記新たに設定されたユーザ名とパスワードによりアクセス可能である請求項７に記載の方法。

【請求項９】　有効なユーザ名とパスワードは前記ユーザアカウントにアクセスするために供給されなければならない請求項８に記載の方法。

【請求項１０】　データネットワーク上で、複数の貧弱なクライアントデバイスのために、プロクシサーバデバイスにおける集中証明書を管理するための装置であって、該装置は、

自由証明書を生成するための証明書管理モジュールと、

上の閾値に達するまで、該証明書管理モジュールからの該自由証明書を格納するための、該証明書管理モジュールと接続された自由証明書データベースと、ユーザアカウントデータベースと、

前記自由証明書データベースにおける前記自由証明書のうちの一つを、新たに活性化された貧弱なクライアントデバイスと関連付けられた該ユーザアカウントデータベースにおける新たなユーザアカウントに関連付けるための証明書割り当てモジュールとを有し、

前記ユーザアカウントデータベースは前記プロクシサーバデバイスによりアクセス可能であり、該ユーザアカウントデータベースは複数のユーザアカウントを有し、各貧弱なクライアントデバイスは該ユーザアカウントのうちの一つと関係付けられ、該ユーザアカウントは該ユーザアカウントに割り当てられたデバイスＩＤ及び証明書リストを有することを特徴とする装置。

【請求項１１】　前記証明書管理モジュールは、

前記証明書割り当てモジュールと通信する証明書エンジンと、

新たな証明書のための一意の名前を生成する名前生成器と、

該新たな証明書のための秘密キー及び公開キーを生成す

るキーペア生成器と、
該新たな証明書のために認証局と交信する証明書要求モジュールとを有し、該証明書要求は前記公開キーと前記一意の名前を有する請求項１０に記載の装置。
【請求項１２】　前記名前生成器は、タイムスタンプと加入者ＩＤとを結合させる識別名生成器を有する請求項１１に記載の装置。
【請求項１３】　前記証明書管理モジュールは、証明書更新要求を受信すると前記自由証明書データベースを更新する請求項１２に記載の装置。
【請求項１４】　前記証明書更新要求は証明書取り消しリストを有する請求項１３に記載の装置。
【請求項１５】　前記証明書更新要求は更に挿入／削除クエリーを有する請求項１４に記載の装置。
【請求項１６】　前記装置は、
前記プロクシサーバデバイスと接続したコンピュータネットワークと、
該コンピュータネットワークに接続されたクライアントコンピュータとを更に有し、該クライアントコンピュータは前記ユーザアカウントデータベースにおけるユーザアカウントにアクセスすることが可能である請求項１０に記載の装置。
【請求項１７】　データネットワークを介してプロクシサーバと接続された複数の貧弱なクライアントデバイスのための、該プロクシサーバデバイスにおける集中証明書を管理する方法であって、
該方法は前記プロクシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記貧弱なクライアントデバイスは該ユーザアカウントのうちの一つと関連付けられ、各ユーザアカウントは、デバイスＩＤ、該ユーザアカウントに割り当てられた公開及び秘密キーのリスト、及び該ユーザアカウントに割り当てられた少なくとも一つの証明書を有し、
該方法は、第１の貧弱なクライアントデバイスと関連した第１のユーザアカウントに割り当てられた第１の証明書を使用して第１の貧弱なクライアントデバイスから、前記プロクシサーバデバイスに接続された安全なサーバにアクセスすることを特徴とする方法。
【請求項１８】　前記プロクシサーバによりアクセス可能な自由証明書データベースを保持し、該自由証明書データベースは認証局（ＣＡ）により発行される複数の自由証明書を有し、各自由証明書は対応する公開キーと対応する秘密キーを有する請求項１７に記載の方法。
【請求項１９】　前記プロクシサーバにおける証明書データベースの保持は、
前記証明書データベースにおける自由証明書の数が下の閾値より低い場合に証明書要求を受け、
新たな証明書を生成し、該新たな証明書の生成は、
該新たな証明書のための識別名を生成し、

該新たな証明書のための新たな公開キー及び新たな秘密キーを生成し、
前記証明書要求をＣＡに送信し、該証明書要求は生成された該新たな公開キーを有し、
ＣＡにより署名された新たな証明書を受信し、
前記自由証明書データベースに前記新たな証明書を置くことからなる請求項１８に記載の方法。
【請求項２０】　前記ユーザアカウントデータベースの保持は、
新たな貧弱なクライアントデバイスが活性化されると、前記自由証明書データベースから前記自由証明書のうちの一つを取得し、
新たなデバイスＩＤ及び新たな加入者ＩＤを有する新たなユーザアカウントを構築し、
取得した前記自由証明書、対応する秘密キー及び公開キーを前記新たなデバイスＩＤを有する前記新たなユーザアカウントと関連付けることからなる請求項１７に記載の方法。
【請求項２１】　証明書更新要求を受信すると前記自由証明書データベースにおける前記自由証明書を更新することを含む請求項１８に記載の方法。
【請求項２２】　前記証明書更新要求を受信すると自由証明書データベースにおける自由証明書を更新することは、
該証明書更新要求が証明書取り消しリストである場合に、前記自由証明書データベースから無効な証明書を取り除くことからなる請求項２１に記載の方法。
【請求項２３】　前記証明書更新要求を受信すると自由証明書データベースにおける自由証明書を更新することは、
前記証明書更新要求における挿入／削除クエリーに応じて前記自由証明書データベースから証明書を削除することからなる請求項２１に記載の方法。
【請求項２４】　有効なデバイスＩＤを有する貧弱なクライアントから新たに設定されたユーザ名とパスワードを受信すると、該有効なデバイスＩＤと関連付けられた前記ユーザアカウントデータベースにおけるユーザアカウントを更新することを含む請求項１７に記載の方法。
【請求項２５】　前記ユーザアカウントデータベースにおける前記ユーザアカウントは、インターネットを介して前記プロクシサーバに接続されたコンピュータからアクセスされる請求項１７に記載の方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、データネットワークにおけるサーバコンピュータとクライアントコンピュータ間のデータセキュリティに関し、特に、データネットワーク上の２方向インターラクティブコミュニケーションデバイスのためのデジタル証明書を、プロクシサーバにおいて管理するシステムに関する。モバイルデバイ

ス、セルラ電話、地上線（landline）電話及びインターネット装置コントローラのような２方向インターラクティブコミュニケーションデバイスにおいては、一般的に、コンピューティングパワー、メモリ及びグラフィック表示能力等のコンピューティング資源が限られている。

【０００２】

【従来の技術】インターネットにおける急成長のトレンドは電子商取引である。電子商取引は、次のものをいっしょに取り入れて設計された統合概念である。すなわち、幅広い領域のビジネス支援サービス、商品・製品・カスタマイズされた製品・特注の品及びサービスのための取引支援、注文及び調達支援システム、決済支援システム、管理情報及び統計報告システムであり、これらは全てインターネットを介して実施される。しかしながら、インターネットは、相互接続された世界中のコンピュータ及び電子デバイスの、公共の開かれた国際ネットワークであるということは良く知られている。安全にデータを送受信する能力は、インターネット上で電子商取引を行う上で基本的な要求となっている。その開かれたネットワーク上でビジネス取引を行うために、企業又は組織は、不正から自分自身及び顧客を守るためのアイデンティティー及び信頼を確立する効率的で信頼性の高い方法を持たなければならない。同様にして、顧客は、インターネットに発信するかもしれない個人情報が送信先の企業以外の誰にも読まれることがない、ということに確信を持つ必要がある。

【０００３】２つのに認証された側の間における個人的コミュニケーション又はビジネス取引を保証するために進行している努力の一つは、その２つの身元を、インターネット上を伝送されるデジタル情報を暗号化し署名するために使用され得る電子キーのペアと結びつけるデジタル証明書を使用することである。デジタル証明書は、与えられたキーを使用する権利を持っているという誰かの主張を確かめることを可能とし、それにより、認証されたユーザになりすまそうとして他人が偽のキーを使用することを防止することに寄与する。暗号化と共に使用されることにより、デジタル証明書は、開かれたネットワークを介して取引に参加する全ての参加者の身元を確認することにより、より完全なセキュリティの解決策を提供する。

【０００４】デジタル証明書を使用する現在の構成は、インターネット上で、クライアントコンピュータとサーバコンピュータの２つのコンピュータの間を結び付けることであり、このことは、両コンピュータが物理的にそれぞれ自身の証明書を持ち、証明書を保持するためのメモリスペースが必要であるということを意味する。それらの証明書のうちの１つが無効（期限切れ、取り消し、又は使用不可）になった場合、その無効の証明書を有するコンピュータは、証明書発行機関から新しい証明書を取得し得る。

【０００５】

【発明が解決しようとする課題】しかしながら、その取得プロセスは一般的に時間が数分かかり、多大なコンピューティングパワーを要する。２つのコンピュータ間のコミュニケーションセッションが確立されると、その２つのコンピュータは、相手の証明書を調べることによって互いに認証しあう。認証が成功すると、セッションキーが生成され、その２つのコンピュータ間を交流する全ての情報を暗号化するためのそのセッションキーを使用して、コミュニケーションセッションが開始される。その認証プロセスはまた多大なコンピューティングパワーを要する。

【０００６】クライアントコンピュータが、モバイルコンピューティングデバイス、セルラ電話、地上線電話又はインターネット装置コントローラのような小さな２方向コミュニケーションデバイスの場合、上記の構成は適用し難い。ポータビリティと移動性を増加させるために、そのような２方向コミュニケーションデバイスの大部分は、小さく、軽く、消費電力を小さく、そしてできるだけ安価に設計される。そのような設計は、しばしば貧弱な（thin）設計と考えられ、それによりコンピューティングパワーは非常に限られ、典型的には、それは典型的なデスクトップ又はポータブルコンピュータにおいて備えられているものの１％以下であり、そのメモリ容量は一般的に２５０キロバイト以下である。そのことは、その貧弱なクライアントデバイスは多数の証明書を格納するための余計なメモリ容量を持たず、保持している証明書の１つが無効になった場合にリアルタイムで新しい証明書を取得するために必要なコンピューティングパワーを持たないということを意味する。従って、効率的に証明書を管理する機構をその貧弱なクライアントに提供する大きな必要性がある。

【０００７】

【課題を解決するための手段】上記の目的を達成するために本発明は次のように構成される。本発明は、データネットワークを介してプロクシサーバと接続された複数の貧弱（thin）なクライアントデバイスのための、該プロクシサーバデバイスにおける集中証明書を管理するための方法であって、その方法は前記プロクシサーバによりアクセス可能な自由証明書データベースを保持し、該自由証明書データベースは認証局（ＣＡ）によって発行された複数の自由証明書を有し、各自由証明書は対応する公開キー及び対応する秘密キーを有し、また、その方法は前記プロクシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記各貧弱なクライアントデバイスは複数のユーザアカウントのうちの一つと関連付けられ、各ユーザアカウントは該ユーザアカウントに割り当てられたデバイスＩＤ、公開及

び秘密キーのリスト、及び、該ユーザアカウントに割り当てられた証明書のリストを有し、更にその方法は前記証明書データベースから取り出された少なくとも一つの証明書を前記ユーザアカウントデータベースにおける各ユーザアカウントに加えることからなる。

【０００８】また、上記目的を達成するために次のような構成としてもよい。本発明は、データネットワーク上で、複数の貧弱なクライアントデバイスのために、プロクシサーバデバイスにおける集中証明書を管理するための装置であって、該装置は、自由証明書を生成するための証明書管理モジュールと、上の閾値に達するまで、該証明書管理モジュールからの該自由証明書を格納するための、該証明書管理モジュールと接続された自由証明書データベースと、ユーザアカウントデータベースと、前記自由証明書データベースにおける前記自由証明書のうちの一つを、新たに活性化された貧弱なクライアントデバイスと関連付けられた該ユーザアカウントデータベースにおける新たなユーザアカウントに関連付けるための証明書割り当てモジュールとを有し、前記ユーザアカウントデータベースは前記プロクシサーバデバイスによりアクセス可能であり、該ユーザアカウントデータベースは複数のユーザアカウントを有し、各貧弱なクライアントデバイスは該ユーザアカウントのうちの一つと関係付けられ、該ユーザアカウントは該ユーザアカウントに割り当てられたデバイスＩＤ及び証明書リストを有する。

【０００９】更に、上記目的を達成するために次のような構成としてもよい。本発明は、データネットワークを介してプロクシサーバと接続された複数の貧弱なクライアントデバイスのための、該プロクシサーバデバイスにおける集中証明書を管理する方法であって、その方法は前記プロクシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記貧弱なクライアントデバイスは該ユーザアカウントのうちの一つと関連付けられ、各ユーザアカウントは、デバイスＩＤ、該ユーザアカウントに割り当てられた公開及び秘密キーのリスト、及び該ユーザアカウントに割り当てられた少なくとも一つの証明書を有し、また、その方法は第１の貧弱なクライアントデバイスと関連した第１のユーザアカウントに割り当てられた第１の証明書を使用して第１の貧弱なクライアントデバイスから、前記プロクシサーバデバイスに接続された安全なサーバにアクセスすることからなる。

【００１０】

【発明の実施の形態】本発明のこれら及び他の特徴、側面、及び利点は、以下の説明、請求項、及び添付図面に関してより良く理解される。

（表記及び述語）下記の本発明の詳細説明において、多くの具体的詳細は、本発明を完全に理解するために述べられる。しかし、本発明がこれらの具体的詳細なしに実施され得ることは当業者にとって明らかになる。次に、良く知られた方法、手順、コンポーネント、及び回路は、本発明の側面をあいまいにすることを回避するために、詳細には説明されない。

【００１１】下記の本発明の詳細な説明は、ほとんど、手順、ステップ、論理ブロック、処理、及び、ネットワークに接続したデータ処理デバイスに似ている他の記号表現に関して行なわれる。これらの処理の説明及び表現は、当業者が他の当業者にその仕事の内容を最も効率的に伝達するために使用する手段である。本発明は、データネットワークにおける２方向インターラクティブコミュニケーションデバイスのための集中証明書管理システムである。以下で詳細に説明される方法は、所望の結果に到達するプロセス又はステップの自己一貫したシーケンスである。これらのステップ又はプロセスは、物理量の物理操作を必要とするものである。通常、必要ではないが、これらの量は、コンピュータシステム又は電気的コンピューティングデバイスにおいて格納され、送信され、結合され、比較され、表示され、その他の操作がされ得る電気信号の形をとり得る。これらの信号を、ビット、値、エレメント、シンボル、オペレーション、メッセージ、項目、数等と称することは、主に一般的に使用されるという理由で、ときどき便利であることがわかる。これらの同様な用語の全ては適切な物理量と関連し、これらの量に適用される単なる便利なラベルである、ということは覚えておくべきである。下記の説明から明らかなように、他に特別に述べられていなければ、本発明を通して、”処理”又は”コンピューティング”又は”確証すること”又は”表示すること”等のような用語を使用した議論は、コンピューティングデバイスのレジスタ及びメモリの中の物理量として表されるデータを、同様にコンピューティングデバイス又は他の電気的デバイスの中の物理量として表される他のデータへ、操作、変換する、コンピューティングデバイスの動作及び処理を意味しているということが認識される。

【００１２】（デジタル証明書の紹介）時々、デジタルＩＤ又はセキュリティ証明書と称されるデジタル証明書又は証明書は、しばしばテキストファイルとして格納され、開かれたデータネットワーク上で２つの当事者間に安全なコネクションを確立するためのセキュアソケットレイヤ（ＳＳＬ）プロトコルによって使用される情報である。最も単純な形において、証明書は公開鍵と名前を含んでいる。良く使用されるものとして、証明書はまた、期限の日付、証明書を発行した認証局の名前、シリアル番号、及びおそらく他の情報を含む。最も重要なことは、証明書が証明書発行者のデジタル署名を含むということである。デジタル署名とは、証明書の内容を確かめるために使用され得る暗号化された”指紋”である。

【００１３】デジタル証明書、もしくは単に証明書は、認証局（ＣＡ）によって発行され、ＣＡの秘密キーを用

いて署名される。デジタルＩＤとして最も広く受け入れられているフォーマットは、国際標準ＣＣＩＴＴ　Ｘ．５０９により定義されている。従って、証明書は、ＣＣＩＴＴ　Ｘ．５０９に準拠したアプリケーションにより読み書きされる。デジタル証明書は、公開キーと秘密キーという関連する２つのキーのペアに基く公開鍵暗号技術を使用している。公開鍵暗号においては、公開キーは、そのキーペアの保持者と通信を欲する者には誰にでも取得可能である。公開キーは、秘密キーにより署名されたメッセージを確認するか、又は、秘密キーを使うことによってのみ復号化され得るメッセージを暗号化するために使用され得る。このようにして暗号化されたメッセージの安全性は、秘密キーの安全性による。秘密キーは不正使用から保護されなければならない。

【００１４】証明書におけるキーのペアは、ユーザ名及び他の識別情報と結びつけられている。カリフォルニアのNetscape Communication Inc.のネットスケープナビゲータ、又は、ワシントンのMicrosoft Corporation のインターネットエクスプローラのようなＨＴＭＬブラウザにインストールされると、その証明書は、コンタクトされたサイトが調べることができる電子信任状として機能する。これは、時々、デジタル証明書が、会員権が必要な情報やサービス、又は、特定のユーザへのアクセスを制限するためのパスワードダイアログと置き換わることを可能とする。例えば、誰かがメッセージを業者のウェブサイトに送る場合、その人はそのメッセージに署名し、そのメッセージが確かにその人から送られたものであるということをそのメッセージの受取人に確信させるためにデジタルＩＤを含める。その業者がデジタル的に署名されたメッセージを受信すると、その署名者のデジタルＩＤは、偽造や不正な表現が起こっていないことを決定するために確かめられる。一般的に、一旦ユーザが証明書を取得すると、そのユーザは、その証明書を自動的に使用するために、セキュリティ機能付きウェブ又は電子メールアプリケーションを設定することができる。図１はクライアントと業者サーバ間でデジタルＩＤを使用する認証プロセスを示す。

【００１５】認証の最も安全な使用方法は、署名された各メッセージに１つかそれ以上の証明書を同封することを含む。そのメッセージの受信者は、認証局の公開キーを使用して証明書を確認し、送信者の公開キーであると確信し、そのメッセージの署名を確認する。メッセージには２つ又はそれ以上の証明書が同封されることがあり、階層的チェーンをなし、そこでは１つの証明書が前の証明書の確実性を保証する。証明書階層の終端はトップレベルの認証局であり、そこは他のいかなる認証局から認証なしで信頼される。トップレベル認証局の公開キーは、例えば、広く発行されることによって、独立して知られていなければならない。言い換えれば、受信者に会社名が知られている送信者は、ただ１つの証明書（その会社により発行されたもの）を同封すればよいが、会社名が受信者に知られていない送信者は２つ又はそれ以上の証明書を同封する必要があり得る。セキュリティーを高度にするために、チェーンにおける最も高いレベルの発行者が受信者に良く知られるのにちょうど十分の証明書チェーンを同封することは一般的な方法である。多数の受信者がいる場合、各受信者が必要とするものをカバーするために十分な証明書を含めねばならない。

（好ましい実施態様）ここで、図面を参照するにあたり、図面を通して同一の数字は同一の部分を示している。図２は、本発明の実施され得るデータネットワーク１００を示す。データネットワーク１００は、一般的にはワイヤレスネットワークと称されるエアネット（airnet）１０２、一般的には地上線（landline）ネットワークであるランドネット(landnet) １０４からなり、各々はそこを通るデータ伝送のためのコミュニケーション媒体である。エアネット(airnet)１０２では空中を介してデータ伝送が行われ、エアネット１０２はＡＴ＆Ｔ又はＧＴＥのようなキャリアによって制御、運営されているので、ときどきキャリアネットワークと称される。各キャリアは、エアネット１０２において、ＣＤＰＤ、ＣＤＭＡ、ＧＳＭ及びＴＤＭＡのような独自の通信スキームを有し得る。ランドネット１０４又はインターネットは、これらはここでは交互に用いられ、インターネット、イントラネット、又は他のプライベートネットワークであり得る。１０６により参照されるものは、１つのモバイルデバイス、セルラ電話、地上線電話若しくはインターネット装置コントローラであり得るモバイルデバイスのうちの１つであり、アンテナ１０８を介してエアネット１０２と通信できる。エアネット１０２は複数の２方向通信デバイスの通信を同時に運ぶが、ただ１つのモバイルデバイス１０６のみが図に示されている、ということは一般的に理解されることである。

【００１６】同様に、複数のデスクトップパーソナルコンピュータ（ＰＣｓ）１１０及び複数のサーバコンピュータ１１２がインターネット１０４に接続されているが、それぞれただ１つの例のみが図に示されている。図に示すように、ＰＣ１１０は、ＮＥＣ　Technologies Inc.　のパーソナルコンピュータＳＰＬ３００であり得、ハイパーテキストマークアップ言語（ＨＴＭＬ）Ｗｅｂブラウザが載っており、そのブラウザは、インターネット１０４を介し、ハイパーテキストトランスファープロトコル（ＨＴＴＰ）を使用して、Ｓｕｎ　Microsystems Inc.　のワークステーションであり得るｗｅｂサーバに格納された情報にアクセスする。ＰＣ１１０はｗｅｂサーバにもなるためのアクセス可能な情報を格納し得るということは当業者には理解される。インターネット１０４及びエアネット１０２の間に、その間でデータを通信するプロクシサーバコンピュータ１１４がある。プロクシサーバコンピュータ１１４は、また、リンクサ

ーバ又はゲートウェイサーバコンピュータと称され、ワークステーション又はパーソナルコンピュータであり得、マッピング又は翻訳の機能を実行する。例えば、１つのプロトコルから他への通信プロトコルマッピングを実行し得、それゆえ、モバイルデバイス１０６は、サーバ１１２又はＰＣ１１０のそれぞれのうちの１つと通信できる。

【００１７】インターネット１０４で使用される１つのよく知られた通信プロトコルは、ハイパーテキストトランスポートプロトコル（ＨＴＴＰ）又はＨＴＴＰのセキュアバージョンであるＨＴＴＰＳであり、ＴＣＰ上で動作し、サーバ１１４における、良く知られたハイパーテキストマークアップ言語ウェブブラウザもしくはＨＴＭＬウェブブラウザのウェブサーバ１１２への接続を制御し、その間での情報の交換を制御する。ＨＴＴＰＳは、ＨＴＭＬブラウザとウェブサーバ間の安全で認証された通信において最も使用されるＳＳＬをサポートしている。ＨＴＭＬブラウザにおいて良く用いられる表示は、ユニバーサルリソースロケータ若しくはＵＲＬの前に"https"を使用するものであり、これはＳＳＬコネクションが確立されることを示している。ＳＳＬコネクションにおける一方の側、好ましくはサーバ側はその相手側によって認証されなければならない証明書を有する。そして、各側は、それ自身、他の側、又は両側の証明書からの情報を使用して送信するものを暗号化し、目的とする受信者のみがそれを復号化でき、データが確かにそれが示す場所から来たということを他の側が確信でき、そのメッセージが改竄されていない、ということを確かめる。

【００１８】エアネット１０２を介したモバイルデバイス１０６とプロクシサーバ１１４の間の通信プロトコルはハンドヘルドデバイストランスポートプロトコル（ＨＤＴＰ）又はセキュアアップリンクゲートウェイプロトコル（ＳＵＧＰ）であり、好ましくは、ユーザデータグラムプロトコル（ＵＤＰ）上で動作し、モバイルデバイス１０６においてＨＤＭＬウェブブラウザのプロクシサーバ１１４への接続を制御する。ここで、ＨＤＭＬはハンドヘルドデバイスマークアップランゲージを意味する。ＨＤＭＬは、ＨＴＭＬと同様、タグベースのドキュメント言語であり、モバイルデバイス１０６の小さなスクリーンに情報がどのように表示されるかを決めるカード（ｃａｒｄ）において指定されるコマンド又はステートメントのセットからなる。通常、多数のカードがデッキ（ｄｅｃｋ）にまとめられ、そのデッキが、モバイルデバイス１０６とプロクシサーバ１１４の間で交換され得るＨＤＭＬ情報の最小ユニットである。"ＨＤＴＰ Specification"のタイトルのＨＤＴＰの仕様及び"ＨＤＭＬ２．０Language Reference"のタイトルのＨＤＭＬは同封され、全体に参照されている。ＨＤＴＰは、ＨＴＴＰに似たセッションレベルのプロトコルであるが、そのオーバヘッドを負わず、非常にコンピューティングパワーとメモリの限られた貧弱なデバイスにおける使用に高度に最適化されている。更に、ＵＤＰは、ＴＣＰの場合のように情報が交換され得る前にクライアントとサーバ間に確立されるコネクションを要しない。それゆえ、ＵＤＰを使用することは、クライアントとサーバ間でのセッション確立の間、多数のパケットを交換させる必要性を削減させる。トランザクションの間に交換するパケットが非常に少ないことは、非常に限られたコンピューティングパワーとメモリしか持たないモバイルデバイスが効率的にランドラインデバイスとやりとりするために望ましい特徴である。

【００１９】モバイルデバイス１０６はディスプレイスクリーン１１６とキーボードパッド１１８から構成される。モバイル電話１０６における、マイクロコントローラ、ＲＯＭ及びＲＡＭを含むハードウェアコンポーネントは当業者には公知であるので、ハードウェアコンポーネントの詳細はここでは説明しない。スクリーン１１６とキーパッド１１８を使用して、モバイル電話１０６のユーザは、エアネット１０２上をプロクシサーバ１１４とインターラクティブに通信できる。１つの実施形態によれば、コンパイルされリンクされた本発明のプロセスは、クライアントモジュールとしてＲＯＭに格納され、モバイルデバイス１０６をプロクシサーバ１１４と動作させる。キーパッド１１８を使用した所定のキーシーケンスによる活性化により、マイクロコントローラは、ＲＯＭの中のクライアントモジュールを使用して、プロクシサーバ１１４への通信セッション要求を初期化する。通信セッションを確立すると、モバイルデバイス１０６は、典型的には、プロクシサーバ１１４から１つのＨＤＭＬデッキを受信し、ＲＡＭにキャッシュとしてそのデッキを格納する。上述した通り、ＨＤＭＬデッキは１つ又はそれ以上のカードからなり、各カードは、ディスプレイスクリーン１１６上にスクリーンディスプレイを生成するために要求される情報を含む。カードデッキの中のカードの数は、モバイルデバイス及びエアネットネットワーク１０２におけるリソースの効率的な使用を促進するために選択される。一般的に、カードのうちの一つは選択カードであり、頻繁に訪れるウェブサイトのシーケンスを示し、ユーザに一つを選択させ、安全で認証された通信セッションがプロクシサーバとで確立する。このような通信セッションを確立させるために証明書を使用するプロセスは以下で説明される。

【００２０】図３には、データネットワークにおける他の部分又はコンポーネントと相互に作用する本発明が示されている。エアネット１０２に接続される複数のモバイルデバイスの３つの表現が３０２、３０４及び３０６により参照され、同様に、ランドネット１０４に接続される複数の地上線デバイスの３つの表現が３１０、３１２及び３１４により参照される。図２におけるプロクシ

サーバ１１４であり得るプロクシサーバデバイス１２８は、エアネット１０２をランドネット１０４に接続する。従って、どのようなモバイルデバイスも、エアネット１０２、プロクシサーバ１１４及びランドネット１０４を介して地上線デバイスと通信できる。本発明の説明を容易にするために、モバイルデバイス３０２とリンクサーバ１１４の内部の構成がそれぞれ示されている。他のプロセス及びハードウェアは当業者には公知であるので、わかりやすさのためにそれらは詳細には示さない。

【００２１】モバイルデバイス３０２のような各モバイルデバイスにはデバイスＩＤ３１６が割り当てられる。デバイスＩＤ３１６は電話番号や、ＩＰアドレスとポート番号の組み合わせ、例えば、204.163.165.132:01905であり得る。ここで、204.163.165.132 がＩＰアドレスであり、 01905がポート番号である。デバイスＩＤ３１６は更に加入者ＩＤ３１８と関連し、加入者ＩＤ３１８は、プロクシサーバ１１４においてユーザアカウント３２４を確立させることによってモバイルデバイス３０２を活性化させる手順の一部としてプロクシサーバ１１４の中でキャリアにより認証される。加入者ＩＤ３１８は、例えば、ＡＴ＆Ｔワイヤレスサービスによる861234567-10900 __pn.mobile.att.net の形をとり得る。加入者ＩＤ３１８はモバイルデバイス３０２の一意の識別子である。言い換えれば、各モバイルデバイス３０２、３０４及び３０６は、プロクシサーバ１１４における各ユーザアカウントを示す加入者ＩＤに対応するそれぞれの一意のデバイスＩＤを有する。以下の説明は、モバイルデバイス３０２及び関連するアカウント３２４に基き、その説明はプロクシサーバ１１４と同時に通信する複数のモバイルデバイスに同様に適用されることは当業者にとって認識されるものである。

【００２２】アカウント３２４は、デバイスＩＤ３１６又は加入者ＩＤ３１８により指し示され、ＵＲＬのようなアドレス識別子により識別されるものであり、ユーザ情報３２２、証明書リスト３２０及び秘密キーリスト３２６からなるデータ構造である。ここで、ユーザ情報３２２はアカウント構成と、ユーザ名とパスワードのような他のアカウントに関連する情報を含む。アカウントのＵＲＬは、例えばwww.att.com/Pocketnet の形をとり得、これは、エアネット１０２がＡＴ＆Ｔワイヤレスサービスにより運用されていることを示している。証明書リスト３２０は、１つ又はそれ以上のＣＡにより発行された指定された証明書のリストを含むか又は指し示し、秘密キーリスト３２６はキーのリストを含み、それぞれのキーは証明書リスト３２０における各証明書に対応する。証明書リスト３２０における全ての証明書は、排他的に特定のアカウントと関連している。一般的に、プロクシサーバ１１４は多数のそのようなユーザアカウントをデータベース３２８に保持し、各ユーザアカウントは、同一のキャリアに加入しプロクシサーバ１１４によってサービスを受けているそれぞれのモバイルデバイスと関連している。証明書はアカウントとそれぞれ関連しているため、１つのアカウントにおける証明書は他のアカウントにおける証明書と異なることは認識され得る。

【００２３】ＣＡから証明書を取得し、秘密キーと公開キーのキーの組を生成するためには、通常のフルパワーデスクトップコンピューターにおいて著しく長い時間がかかる。モバイルデバイスを使用して証明書を取得する時間の長さを最小にするために、（証明書管理モジュール（certificate manager module））ＣＭＭ３４２が証明書データベースを、好ましくはデータベース３２８に保持し、１つ又は異なるＣＡからの、自由証明書と称される、指定されていないが発行された証明書のリストを保持する。証明書を必要とするウェブサーバにアクセスするために１つ又はそれ以上の証明書を必要とするモバイルデバイスを活性化するために、ユーザアカウントが作成される度に、証明書要求信号（certRequest ）が、証明書データベースから必要な証明書を取得するために、ＣＭＭ３４２に送られる。証明書データベースから取得された証明書を受信すると、ＣＭＭ３４２は、デバイスＩＤ３１６と他のアカウント情報を付加することによって、証明書を特定のアカウントに割り当てる。それゆえ、取得された証明書は、特定のアカウントと結び付けられ、証明書リスト３２０に置かれる。その間に、ＣＭＭは証明書データベースの中で使用可能な自由証明書の数を調べ、その数が閾値と称されるある値以下、例えば２００以下であれば、ＣＭＭはＨＴＴＰモジュール３３０を呼び、ランドネット１０４を介して適切なＣＡとコネクションを確立し、閾値に達するまで新しい自由証明書を取得して証明書データベースを満たす。そのようにして、証明書データベースには、新しいアカウントにすぐに使用できる自由証明書を救急するために、常に十分な自由証明書が使用可能となっている。

【００２４】普通は十分なコンピューティングパワーを有するローカルデバイスにおいて証明書を取得する従来の方法とは異なり、更に、ローカルデバイスにユーザアカウントを物理的に格納する従来の方法と異なり、本発明は、証明書を非同期に取得するタスクを実行するためにプロクシサーバのコンピューティングパワーを使用しており、ユーザアカウントにおける証明書をプロクシサーバに保持している、ということは当業者にはここで認識され得る。全てのユーザのためにプロクシサーバにおいて証明書を管理することは、クライアントが、追加のコンピューティングパワー及びメモリを要求することなく、安全なウェブサイトにアクセスすることを可能する。他の利点は下記の説明において認識されるだろう。

【００２５】証明書は、証明書及びそれと関連したものを与えられたキーを用いて発行する先のアイデンティティを保証する信頼された中央管理であり得るＣＡによって発行されるものである。例えば、会社又は大学はそれ

自身の従業員や学生に証明書を発行する。ＣＭＭ３４２が自由証明書を取得する先のもの以外に、モバイルデバイス３０２が証明書をＣＡから取得する必要性に対応するために、サーバモジュール３４０は、ユーザに、ランドネット１０４に接続された、例えばＰＣ３１４のようなコンピュータを介してモバイルデバイス３０２に関連するユーザアカウント３２４にログオンさせる。これは、例えばwww.att.com/Pocketnet のようなユーザアカウント３２４のアドレス識別子を使用してユーザアカウント３２４にログオンすることにより達成される。アカウント３２４が許可されたユーザによってアクセスされていることを確認するために、ユーザ名とパスワードのような信用情報の組が要求される。ユーザがhttp://www.att.com/Pocketnetを使用してＰＣ３１４をＵＲＬに接続すると、ＨＴＴＰモジュール３３０を介してサーバモジュール３４０はユーザ名とパスワードを促す。マッチするユーザ名とパスワードの組の入力はそのアカウントにアクセスするための許可として認められる。

【００２６】アカウントに柔軟性とセキュリティを与えるために、ユーザ名とパスワードはユーザにより完全に管理される。モバイルデバイス３０２のユーザは、ＨＤＭＬブラウザを備えたモバイルデバイス３０２を用いてプロクシサーバ１１４におけるデバイスアカウント３２４にアクセスすることができる。アカウントのＵＲＬを知れば、ユーザは、クライアントモジュール３３２にＵＲＬとデバイスＩＤ３１６からなる要求をＵＤＰインターフェース３３６へ送信させるために、所定のキーを押し、ＵＤＰインターフェース３３６は次にＨＤＴＰを使用してプロクシサーバ１１４への通信セッションを確立する。その要求は、プロクシサーバ１１４における対応するＵＤＰインターフェース１２８により受信され、デバイスＩＤが許可されているかどうかを調べるためにサーバモジュール３４０により実行される。そして、プロクシサーバ１１４は、モバイルデバイス３０２に送信される返答を用いて、ユーザ名とパスワードの要求を確認する。その返答は、アカウントへのアクセスを許可するためのユーザ名とパスワードの組を、ユーザから要求するのでなく、実際は、アカウントへのアクセスの許可は、モバイルデバイス３０２からの要求におけるデバイスＩＤ３１６とプロクシサーバ１１４におけるアカウント３２０の格納されたデバイスＩＤとの一致により得られたのである。

【００２７】そのかわり、その返答は、新しいユーザ名とパスワードの組を入力することによりユーザがアカウントを自分で設定することを許容する。アカウント３２０が新しいユーザ名とパスワードの組を受信すると、アカウント、すなわちユーザ情報３２２は更新される。自分での設定の手順の後、ユーザは、好ましくは十分なコンピューティングパワーを有し、より慣れたＨＴＭＬブラウザを備えているＰＣ３１４を使用し、ＨＴＴＰ及びアカウントへのＵＲＬを使用して通信セッションを確立し得る。新たに設定されたユーザ名とパスワードは、促されたときにＰＣ３１４に入力され、ＨＴＴＰを使用してプロクシサーバ１１４へパケットフォーマットで送られる。プロクシサーバ１１４では、ＨＴＴＰサーバ３３０がユーザ名とパスワードを抽出し、サーバモジュール３４０は、メモリの中のユーザ情報３２２を用いて承認チェックを実行する。入力されたユーザ名とパスワードが一致すれば、承認され、ユーザ又はＰＣ３１４はアカウント３２４にアクセスすることを許可される。ここで、ユーザは特定のＣＡから証明書を要求でき、証明書リスト３２０とキーリスト３２６を更新できる。ＨＴＭＬブラウザを使用してＣＡから証明書を取得するプロセスは、当業者には公知であり、従って、ここでは説明されない。ＣＭＭにより提供される機能に加えて、自分で設定する機能により、モバイルデバイス３０２に指定された全ての証明書を保つためにプロクシサーバを信頼しながら、ユーザがその必要に応じて証明書を作ることができる。

【００２８】図３、図４及び図５には、モバイルデバイス３０２のユーザがユーザ特定のＣＡから証明書を要求する例が示されている。所定のキーが押された後、モバイルデバイス３０２はＨＤＴＰを使用し、モバイルデバイス：30286123456-10900 __pn.mobile.xyz.net を示すアカウントのＵＲＬを使用してプロクシサーバ１１４に接続するための要求を行う。デバイスＩＤ86123456-10900はその要求から抽出され、同じデバイスＩＤ86123456-10900により示されるアカウント３２４があることを確認する。確認において、モバイルデバイス３０２のユーザはユーザ名とパスワードの組を促される。ユーザ名とパスワードはモバイルデバイス３０２がアカウント３２４にアクセスするために必要な情報でなく、ユーザがユーザ名とパスワードを管理する許可を得るものである、ということは説明した。ユーザが新たなユーザ名とパスワードを入力しなければ、ユーザアカウント３２４におけるユーザ名とパスワードは同一のままである。ユーザが新たなユーザ名とパスワード、例えばユーザ名”スミス”、パスワード”１２３４５６”の組を入力すると、アカウント３２４は新しいユーザ名とパスワードに更新される。ここで、ユーザはアカウント３２４を操作するために、ランドネット１０４の中のいかなるコンピュータにも行くことができる。ＰＣ３１４は、ユーザにより効率的にアカウント３２４を操作させるための全グラフィカルユーザインターフェースを提供するＨＴＭＬブラウザを備えている。ＰＣ３１４は、サーバモジュール３４０の中のゲートウェイ３５４のＵＲＬ、例えばmobile.xyz.netを使用して、プロクシサーバ１１４における全てのユーザアカウントへのＨＴＴＰコネクションを確立する。ユーザは、ユーザ名とパスワードの組をＰＣ３１４にて促される。ユーザは、ゲートウェイ３５４を通

過するために、ユーザ名として”スミス”、パスワードとして”１２３４５６”を入力しなければならない。入力されたユーザ名とパスワードを受信すると、ゲートウェイ３５４はアカウント３２４におけるものと比較する。ミスマッチの場合、ＰＣ又はユーザはアカウント３２４へのアクセスを許可されない。入力されたユーザ名とパスワードがアカウント３２４におけるものと一致すれば、ゲートウェイ３５４はＰＣ３１４へ許可を与える。ＰＣ３１４のユーザは、特定のＣＡ３５８のＵＲＬを与えることによって、特定のＣＡ３５８から特定の証明書を要求するためにＨＴＭＬブラウザを使用して、使用するモバイルデバイス３０２のためアカウントに証明書を置くことができる。

【００２９】本発明の一つの実施の形態によれば、証明書リストは証明書テーブル３６８へのポインタとして実装され得る。図５に示すように、ポインタを使用することによって証明書リストの柔軟な容量を提供できる、ということは当業者にとって認識され得ることである。証明書インデックス３７０は、全ての証明書と、証明書インデックス３７０において特別に要求された証明書のためのＵＲＬと関連する対応するＵＲＬリスト３７２を格納する空間を提供する。あるＣＡから証明書を受け入れるいくつかのサービスウェブサイトがある。例えば、www.financial.com により識別される金融のウェブサイトはＣＡ　Ｓ１により署名される証明書のみをとる。自分でアカウントを設定することによって、ユーザは、ＣＡ　Ｓ１からの証明書を特に要求でき、その証明書を証明書テーブル３６８に置くことができる。後の使用において、モバイルデバイス３０２はwww.financial.com へのコネクションを確立するための要求を送信する。www.financial.com からなる要求がプロクシサーバデバイス１１４にて受信されると、ＵＲＬは対応する証明書、この場合はＣＡ　Ｓ１による証明書を取得するために使用される。ＣＡ　Ｓ１による証明書を用いて、モバイルデバイス３０２はwww.financial.com によって識別されるウェブへアクセスできる。一般的に、ＣＭＭ３４２によって取得された証明書３８２は、多くのウェブサイトで受け入れられる一般的なものものであり、特定のＵＲＬと関連はしていない。言い換えれば、証明書テーブル３６８は、ユーザによって特別に要求される、３７６、３７８及び３８０によって参照されるような、多くの特定の証明書と、ＣＭＭ３４２によって自動的に取得される、３８２によって参照されるような、１つ又はそれ以上の一般的証明書とを有し得る。

【００３０】図６はＣＭＭ３４２における種々のコンポーネントのブロック図である。上述したように、ＣＭＭは証明書データベースに固定数の自由証明書を保持し、証明書データベースにおける自由証明書の数が閾値より小さくなるとすぐに、ＣＡからＨＴＴＰサーバ３３０を通して新たな証明書を取得し始める。ＣＭＭ３４２において他のコンポーネントの動作を管理する証明書エンジンが４０２により参照されている。モバイルデバイス３０２が活性化されると、そのアカウントは１つ又はそれ以上の証明書をロードするように要求される。自由証明書が証明書データベースから取得された後、証明書データベースにおける使用可能な自由証明書の数が閾値異以下であることを発見すると、エンジン４０２は、識別名生成器４０４又はＤＮ生成器に、生成される新しい証明書にための一意の式別名を生成させる。

【００３１】識別名は、ＣＣＩＴＴＸ．５０９標準における標準形式の名前である。識別名は１つ又はそれ以上の関連識別名からなり、各関連識別名は１つ又はそれ以上の属性値主張（attribute-value assertion）からなる。各属性値主張は、属性識別子と対応する値情報からなり、例えば、CountryName（国名）＝US、Organization（組織）＝XYZ, Inc 又はOrganizationUnit（組織ユニット）＝XYZ Service Division　である。識別名の使用は、Ｘ．５００ディレクトリツリーにおける要素を識別するためであり、ここで、ディレクトリツリーは、インターネットのための”ホワイトページ”一人、コンピュータ、サービス、及び電子メールアドレスのディレクトリーを実装するために使用されている。そのディレクトリは階層的に構成されている。すなわち、トップには国際組織と国があり、国は州や地方に分けられ、それは更に種々の方法で分けられる。関連識別名は、ディレクトリツリーにおける１つのノードから下位のノードへのパスである。全体の識別名は、ツリーのルートから特定の要素を表す終わりのノードへのパスをトラバースする。ディレクトリのゴールは、インターネットにおける全ての通信要素を一意に命名する基盤を提供することであり、それゆえ、識別名において”識別”される。

【００３２】識別名生成器４０４により生成された識別名が結局ユーザ名と関連付けられていることを確かめるために、識別名プレフィックス生成器４０６が、その識別名のためのプレフィックスを生成する。そのプレフィックスは一般的にタイムスタンプと加入者ＩＤとの結合であり、例えば、８６１７６５２２８－９であり、タイムスタンプがいつ証明書要求がなされたかを示し、加入者ＩＤは、活性化されたときに、モバイルデバイスに割り当てられる。識別名プレフィックス生成器４０６からのプレフィックスを用い、識別名生成器４０４からの識別名は一意でなければならない。言い換えれば、証明書データベースにおける各自由証明書はそれ自身の名前を有し、全ての名前は識別されなければならない。

【００３３】証明書エンジン４０２は、公開キーと秘密キーのペアを生成するためにキーペア生成器４１２、若しくはＫＰ生成器を行使する。供給される基となる情報を用いて生成される公開キーに基いて秘密キーを生成するライブラリ機能のセットを使用することにより、それを行う。産業標準に従うために、キーペア生成器４１２

において使用されるライブラリのセットは、Marine Parkway, Suite 500,Redwood City, CA 94065 に住所を持つRSA Data Security,Inc により供給される。生成されたキーは一般的に、1110101100001...00101 のような、バイナリ数のシーケンスの形をとり、それらを生成するソースを知ることなく重ならない。一意の秘密キー及び公開キーのペアを生成するために、ソースとしての乱数がライブラリのセットに応じて提供されなければならない。乱数を得る多くの方法があることは当業者には理解される。一般的に使用される方法のうちの一つは、ハードコード化され得るノイズのソースから、若しくは、ネットワークトラヒック情報からの一方向ハッシュ関数を通して乱数を生成する方法である。一方向は、一つの方向（進行方向）は反対の方向（逆方向）よりも非常に簡単に実行できることを意味し、これにより、公開キーから秘密キーを得ることができなくなる。そのようなハッシュ関数の一つの例は、それ自身の値をある回数掛け、続いてモジュロ操作を行う。

【００３４】証明書エンジン４０２は、証明書データベースの中の証明書のための新しいエントリを作成し、キーペアからの対応する秘密キーはその新たなエントリに格納され、その間、証明書エンジン４０２は、証明書署名要求若しくはＣＲＳを生成するために、生成された識別名及びキーペア生成器４１２から得られた公開キーを使用する。ＣＳＲはＣＡから証明書を要求するための公開の標準フォーマットである。ＣＳＲは、とりわけ、ＣＡによって証明される公開キーとその公開キーと関連する識別名を含む。ＣＳＲは、標準形式で証明書要求の中にパッケージされたデータのバイナリブロックであり、ＨＴＴＰを使用してＨＴＴＰモジュール３３０を通してＣＡに送信される。

【００３５】証明書要求を受信すると、ＣＡは提供されたその情報を確認し、証明書を署名することにより他の情報とともにユーザの公開キーの正当性を証明する。そして、ＣＡは証明書応答を発行する。そこには署名された証明書又はエラーが含まれ得る。証明書応答がエラーを含む場合、これは、要求された証明書が失敗したことを意味し、新たなプロセスが開始されなければならない。証明書応答がＣＡからかえってきた場合、証明書エンジン４０２は受信した証明書から識別名を抽出し、証明書データベースの中の対応するエントリを、証明書格納ライブラリ４０８を通して更新する。この時点で、そのエントリは、公開キーが埋め込まれた署名された証明書と対応する秘密キーを含む。それは、自由証明書として参照されてきたものである。

【００３６】モバイルデバイス３０２が活性化されると、デバイスのための証明書を生成するための要求が出される。証明書エンジン４０２は証明書データベースから自由証明書を取得し、デバイスＩＤと関連付ける。その関連付けは、device__cert__map __tbl と呼ばれる、好ましくはプロクシサーバ１１４のＲＡＭの中にある分離した一時的なテーブルにエントリを作成することにより実行される。

【００３７】証明書格納（ＣＳ）ライブラリ４０８、又はＣＳライブラリは証明書データベースを管理するために使用され、時々、ＣＡから証明書廃止リストを受信する。証明書廃止リスト（ＣＲＬ）は、予定の期限切れ日前に廃止された証明書のリストである。なぜ証明書が廃止されなければならず、ＣＲＬに置かれるかの理由はいくつかある。例えば、証明書の中で特定されるキーは損なわれたかも知れず、証明書の中で特定されるユーザはもはやキーを使用する権限を持たないかもしれない。より詳細には、キーに関連するユーザ名が ”ＸＹＺ会社の副社長スミス氏”であり、スミス氏がその会社を去った場合、その会社は、彼にそのキーでメッセージに署名させたくないだろう。従って、その会社はＣＲＬを証明書に置く。署名を確認するとき、その署名者の証明書が廃止されていないかを確かめるために関連するＣＲＬをチェックすることができる。このチェックを実行するための時間に見合うだけの価値があるかどうかは、署名された文書の重要性に依る。ＣＲＬは、ＣＡによって保持され、ＣＡにより発行された廃止証明書についての情報を提供する。しかしながら、無効な証明書はいかなる場合も受け入れられないので、ＣＲＬは現在有効な証明書のみをリストしている。廃止された証明書がその元の期限日を過ぎた時、それはＣＲＬから除かれる。ＣＲＬは分散して保持されるが、ＣＲＬの集中貯蔵所があり得、それは、多くの組織からの最新のＣＲＬを含むネットワークサイトである。

【００３８】証明書ライブラリ４０８はそのようなＣＲＬを受信し、ＣＭＭ３４２によって維持されている証明書がリストにあると、証明書エンジン４０２に処理を行うように通知する。ＣＭＭ３４２は固定数の自由証明書を有する証明書データベースを維持する。ＣＭＭ３４２が証明書データベースからの証明書をユーザアカウントに関連付けるとき、その関連付けられた証明書は有効でなければならない。これは、ＣＳライブラリ４０８を通して最初にＣＲＬを調べることによって保証される。証明書データベースから取得された証明書がとにかくＣＲＬ上にあれば、取得された証明書は捨てられ、次の証明書が証明書データベースから取得される。ＣＲＬを用いた取得された証明書の照合調査は、取得された証明書がアカウントに関係付けられる前に、常にＣＳライブラリ４０８の中で実行される。ＣＲＬを用いた照合調査は網羅的比較てあり得ることは当業者には理解される。その照合調査にかかる時間又は計算量は、その全てがモバイルデバイス４１６と非同期でプロクシサーバ１１４内で実施されているので、ＣＲＬの長さにかかわらず、許容できる。

【００３９】本発明の実施の形態によれば、付録のソー

スコードリストがCMMにおける動作を表している。main () 関数が、ＨＴＴＰクライアントベースの要求のサービスをするために必要なスレッドを生成するInitializeという名前の関数により初期化されたTCertEngine オブジェクトとコールを作成する。また、証明書データベースの中の証明書をモニタするスレッドも作る。そのスレッドが作成されると、それは可能なリソースをモニタし、証明書を生成するためにTCertHttpProtoの中のGenerateCertを呼ぶ。このスレッドは、証明書プールのために、データベースの中の新たなエントリを生成するためにTDBCertPool を使用する。

【００４０】GenerateCert関数は、識別名生成器から新たな識別名を取得する。それはまた、キーペア生成器から新しい公開／秘密キーの組を取得する。GenerateCertはＣＳＲを構築するためにこの情報を使用した。そして、それは、THttpCertRequestの中のSendCSR メソッドを使用して、ＨＴＴＰ上でＣＡへの要求を発行する。証明書応答がＣＡから返されると、TDBCertPool を使用して自由プールの中のエントリを更新する。

【００４１】自由証明書がユーザアカウントと関連付けられる必要があると、TcertCreateCallback の中のHandleCreateCertメソッドが呼び出される。そして、そのメソッドはdevice__cert__map __tbl の中の新たなエントリを作成するためにTDBDeviceMapの中の関数を呼ぶ。そして、呼び出した側に応答を返す。再発行スレッド、TCertReissueThreadは証明書を再発行するためにTcertHttpProtoにおけるReissueCert を呼ぶ。それは、自由プールにおける証明書及びデバイスに関連する証明書を廃止するためにTDBCertPool 上のメソッドをTDBDeviceMapへ呼び出す。

【００４２】図７と図８は、本発明の集中証明書管理システムの動作フローチャートを示し、図３、図４、図５と合わせて理解されるべきである。本発明のコンパイルされリンクされたプロセスはプロクシサーバ５０２にロードされ、プロクシサーバ５０２に集中証明書管理をさせる。プロクシサーバは、一般的に十分なコンピューティングパワーとメモリを備えたサーバコンピュータ若しくはデバイスであり、そのデバイスには、そのデバイスに他のコンピューティングデバイスへのサービスをさせるアプリケーションが搭載されており、従って、そのアプリケーションは一般的にサーバと称され、そのデバイス自身はここではサーバデバイスと称される。本発明のコンピューティングデバイスは、モバイルデバイス、セルラ電話及びインターネット装置コントローラであり得る貧弱なデバイスである。

【００４３】５０４において、ＣＭＭ３２４は、好ましくはサーバデバイス５０２におけるローカルの格納ドライバに格納されている証明書データベースを、サーバデバイス５０２に維持させる。証明書データベースは、ＣＡによって署名された、ユーザアカウント又は貧弱なクライアントとまだ関連付けられていない固定数の自由証明書を保持している。データベースにすぐに使用可能な自由証明書を維持することによって、その貧弱なクライアントは、著しい時間遅延なく、コンピューティングパワー及びメモリの追加なく、関連する証明書を取得できる。５０６において、使用可能な自由証明書の数が調べられる。その数が落ちると、新たな証明書を取得するステップが５１０にて開始する。証明書データベースにおける自由証明書の数は、時々、５０８における証明書更新により落ちる、ということは理解されるべきである。ユーザアカウントと関連付けされた証明書が常に有効であることを確認するために、ＣＳライブラリ４０８は、ＣＡ又は良く使用される貯蔵サイトから受信された証明書の更新メッセージに従って、証明書データベースにおける自由証明書をコンスタントに更新する。その証明書のメッセージはＣＲＬ又は挿入／削除クエリから構成され得、ＣＭＭ３２４にいくつかの自由証明書を捨てさせ、それにより自由証明書の数が減少する。いかなる場合でも、ＣＭＭ３２４は、ＣＡから新しい証明書を取得することによって証明書データベースの中の自由証明書のレベルを保とうとする。新しい証明書を取得するプロセスが開始すると、ＣＭＭ３２４は最初に、５１０と５１２において、ＤＮプレフィックス生成器４０６とＤＮ生成器４０４を呼び出すことにより、新たな証明書のための識別名を得、そして、５１４において秘密キーと公開キーの組を生成するためのＫＰ生成器４１２を呼ぶ。証明書要求が、５１６において、生成された識別名と公開キーからなるＣＳＲを含ませるために形成される。５１８において、ＣＭＭ３４２は、ＨＴＴＰサーバ３３０を介してＨＴＴＰを使用してＣＡと通信する。証明書要求を受信すると、ＣＡは、証明書に署名することにより他の情報とともに公開キーの有効性を認証し、ＣＭＭ３４２に証明書応答を返し、署名された証明書が５２０で作成される。署名された証明書は、５２２において、自由証明書として証明書データベースに預けられる。論理的には、自由証明書の数は１づつ増やされ（インクリメントされ）、固定数又は閾値と比較される。インクリメントされた数がまだ閾値以下である場合、新たな証明書を取得するプロセスは５１０から、証明書データベースの中の自由証明書の数が閾値に達するまで繰り返される。

【００４４】その間、ＣＭＭ３４２は５３６にて複数のユーザアカウントを維持し、それらのアカウントは好ましくはそれぞれ１つの貧弱なクライアントに割り当てられる。各アカウントは、そのアカウントに排他的に関連した１つ又はそれ以上の証明書を有する。貧弱なクライアントが、サーバデバイス５０２によりサービスを受けるために活性化されると、新しいユーザアカウントが５３８にてそのために確立される。前述したように、ユーザアカウントはデバイスＩＤ、加入者ＩＤ、ユーザ情

報、証明書リスト、秘密キーリストからなり得る。デバイスＩＤは、サーバデバイス５０２がどの貧弱クライアントデバイスにサービスするかを認識することを助ける情報であり、貧弱情報が活性化すると入力される。ユーザ情報は、貧弱クライアントが必要とするアカウント構成とサービスに関する情報を含む。加入者ＩＤ、証明書リスト及び秘密キーリストは、証明書がそれに関連付けられる時に得られる。５４０において、証明書を取得するための要求が作成される。５４２にて要求を受信すると、ＣＭＭ３４２は証明書データベースから有効な自由証明書を取得し、自由証明書をアカウントに関連付ける。

【００４５】本発明は自分で設定する方法を含む。特に、ユーザは図８におけるステップ５４４に示されるように自分での設定を試み得る。第一に、その試みはアカウントにログオンすることによりユーザアカウントにアクセスすることである。ユーザがデバイスＩＤを有する貧弱なクライアントデバイスを使用してログインする場合、アクセスはすぐに確証される。ユーザがインターネットに接続したＰＣを使用してログインする場合、今度は、ユーザは現在のユーザ名とパスワードを入力しなければならない。アクセスを取得した後、ユーザは、ステップ５７２にてユーザ名及び／又はパスワードを変更し得る。ユーザはそして、５７８において、ユーザ特定のＣＡから証明書を要求するために、貧弱なクライアント又は他のコンピューティングデバイスからアカウントにアクセスできる。

【００４６】サーバデバイス５０２において１つ又はそれ以上の証明書を有するアカウントを構築することによって貧弱なクライアントが活性化した後、その貧弱なクライアントは、安全で認証された通信セッションを、いくつかの安全なウェブサイトと秘密通信を行うために、確立することができる。５５０において、サーバデバイス５０２は、ＵＲＬで識別されたウェブサイトと安全で認証された通信セッションを確立するために、貧弱なクライアントからセッション要求を受信する。サーバデバイス５０２が貧弱なデバイスを認識し、その結果そのような要求を認証するために、そのセッション要求は、貧弱なクライアントのデバイスＩＤからなる。５５２において、デバイスＩＤはセッション要求から抽出され、ユーザアカウントの中のデバイスＩＤと比較される。デバイスＩＤが一致すれば、貧弱なデバイスは認証され、更に、対応するアカウント毎に調べられる。５４４において、一致したアカウントにおける証明書が取得され、ＨＴＴＰＳを使用して所望のウェブサイトに送られるセッション要求に含められる。５５８において、貧弱なクライアントと接続したウェブサイト間の認証はそれぞれ相手の証明祖を調べることによって実行される。各証明書が信頼されると、セッションキーがそこから結果として生じ、その貧弱なクライアントとウェブサイト間で交換される情報を暗号化するために使用される。

【００４７】本発明は、ある詳細の程度をもって、十分に詳細に説明された。本実施の形態の開示は例のみを介してなされ、ステップとともに部分のアレンジメント及び組み合わせにおける多大な変更はクレームされた本発明の精神とスコープから離れることなく行なわれ得る、ということは当業者には理解される。従って、本発明のスコープは、一実施形態の説明よりも添付のクレームにより定義される。

【００４８】（マイクロフィッシュ付録の参照）本発明の開示の一部である付録Ａは、”データネットワークにおける２方向コミュニケーションデバイスのための集中証明書管理システム”という題名の、全部で１８４フレームを有する２シートからなるマイクロフィッシュ付録である。そのマイクロフィッシュ付録は、本発明のワイヤレスデータネットワークにおける２方向コミュニケーションデバイスのための集中証明書管理システムの一実施例のソースコードリストであり、上記においてより完全に説明されている。

【００４９】この特許文書の開示の部分は、著作権保護の対象となり、付録Ａ、Ｂ、及びＣを含むがこれに限定されないマテリアルを含む。著作権保持者は、それが特許商標庁の特許ファイル又は記録にあれば、本特許文書又は特許開示のいかなる者によるファクシミリ複製にも反対理由を持たない。しかしながら、それ以外の場合、著作権保持者は全ての著作権を保持する。

【図面の簡単な説明】

【図１】クライアントデバイスと業者のサーバ間で証明書がどのように使用されるかの模範を示す図である。

【図２】本発明が実施され得る、エアネット及びランドネットからなるモバイルデータネットワークを示す図である。

【図３】データネットワークにおいて他の部分及びコンポーネントと対話する本発明を示す図である。

【図４】モバイルデバイスのユーザが、ユーザの特定するＣＡから証明書を要求する例を示す図である。

【図５】モバイルデバイスのユーザが、ユーザの特定するＣＡから証明書を要求する例を示す図である。

【図６】本発明の認証モジュールにおける種々のコンポーネントのブロック図である。

【図７】データネットワーク上で、貧弱なクライアントのためにサーバデバイスにおいて証明書を管理するためのプロセス及び手順を示す動作フローチャートである。

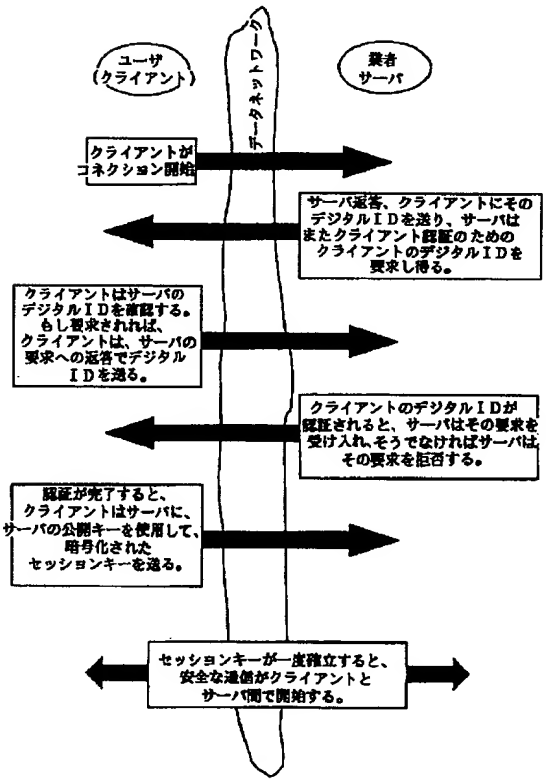
【図８】データネットワーク上で、貧弱なクライアントのためにサーバデバイスにおいて証明書を管理するためのプロセス及び手順を示す動作フローチャートである。

【符号の説明】

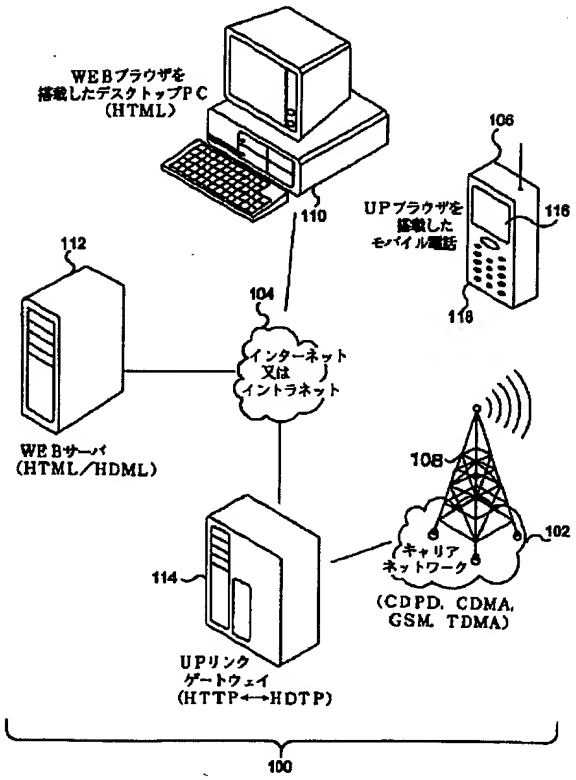
１００　データネットワーク
１０２　エアネット（airnet）
１０４　ランドネット（landnet）

１０８　アンテナ
１０６、３０２、３０４、３０６　モバイルデバイス
１１０　デスクトップパーソナルコンピュータ
１１２　サーバコンピュータ
１１４　プロクシサーバコンピュータ
１１６　ディスプレイスクリーン
１１８　キーボードパッド
１２８　ＵＤＰインターフェース
３１０、３１２、３１４　地上線デバイス
３１６　デバイスＩＤ
３１８　加入者ＩＤ
３２０　証明書リスト
３２２　ユーザ情報
３２４　ユーザアカウント
３２６　秘密キーリスト
３２８　データベース３２８
３３０　ＨＴＴＰモジュール
３３２　クライアントモジュール
３３４　メモリ
３３６　ＵＤＰインターフェース
３４０　サーバモジュール
３４２　ＣＭＭ
３５４　ゲートウェイ
３５６、３５８　ＣＡ
３６８　証明書テーブル
３７０　証明書インデックス
３７２　ＵＲＬリスト
３７４、３７６、３７８、３８０、３８２　証明書
４０２　証明書エンジン
４０４　識別名生成器
４０６　ＤＮプレフィックス
４０８　証明書格納ライブラリ
４１０　証明書要求モジュール
４１２　キーペア生成器
４１４　シード生成器

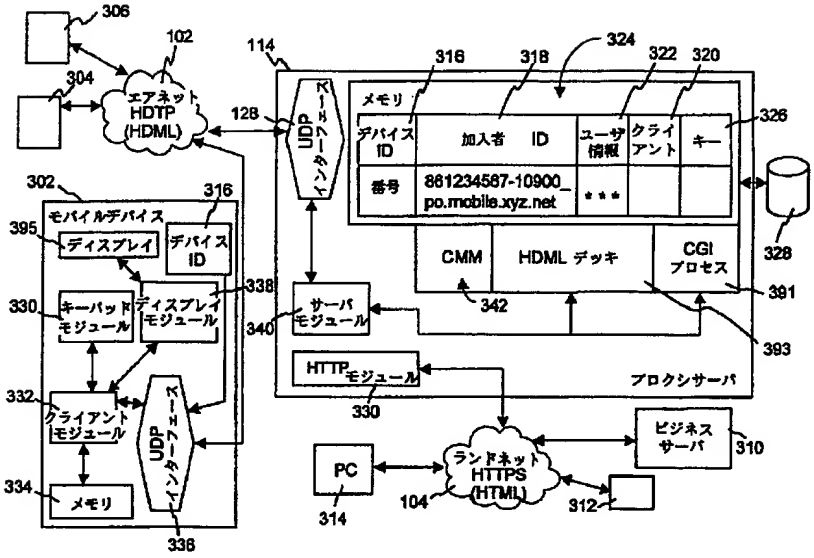
【図１】



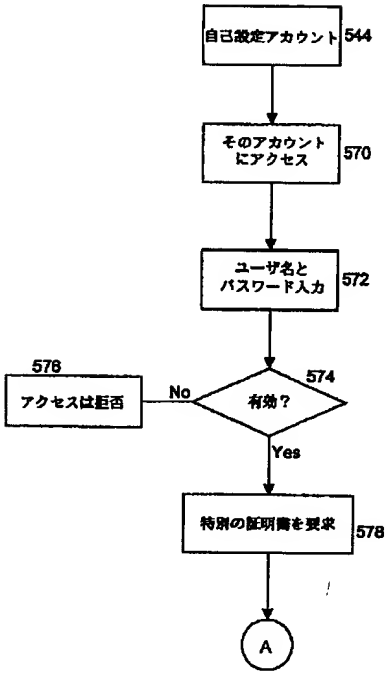
【図２】



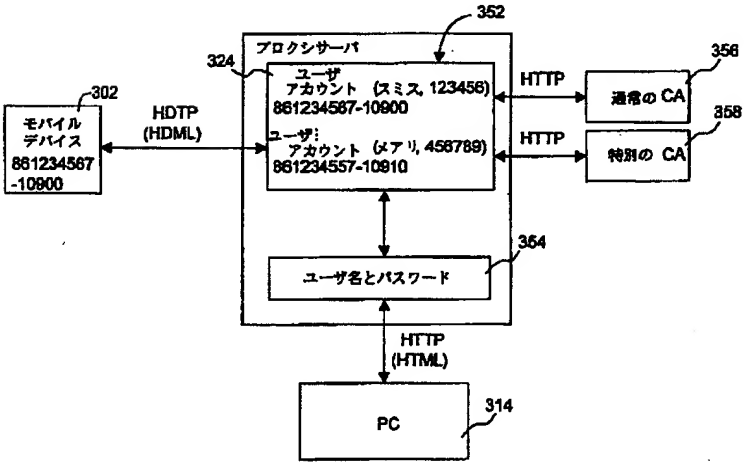
【図３】


306
102
304
エアネット
HDTP
(HDML)
128
114
UDP
インターフェース
316
318
324
322
320
メモリ
デバイス ID
加入者 ID
ユーザ情報
クライアント
キー
326
番号
861234587-10900_
po.mobile.xyz.net
328
CMM
HDML デッキ
CGI プロセス
391
342
393
302
モバイルデバイス
395
ディスプレイ
デバイス ID
330
キーパッド モジュール
ディスプレイ モジュール
338
340
サーバ モジュール
HTTP モジュール
330
プロクシサーバ
332
クライアント モジュール
UDP インターフェース
334
メモリ
336
PC
314
ランドネット
HTTPS
(HTML)
104
312
ビジネス サーバ
310

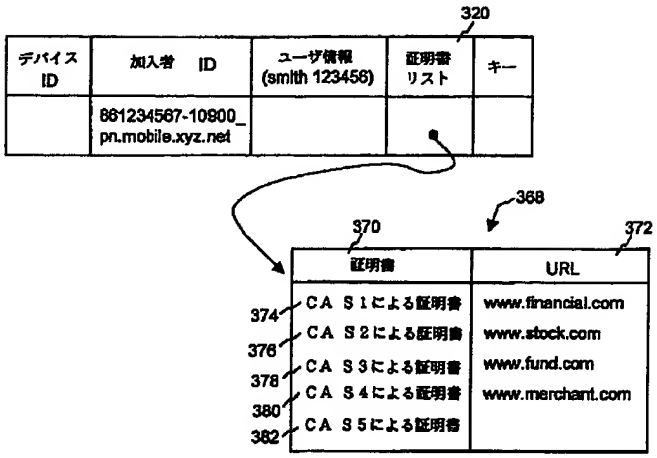
【図８】


自己設定アカウント
544
そのアカウント にアクセス
570
ユーザ名と パスワード入力
572
576
アクセスは拒否
No
574
有効？
Yes
特別の証明書を要求
578
A

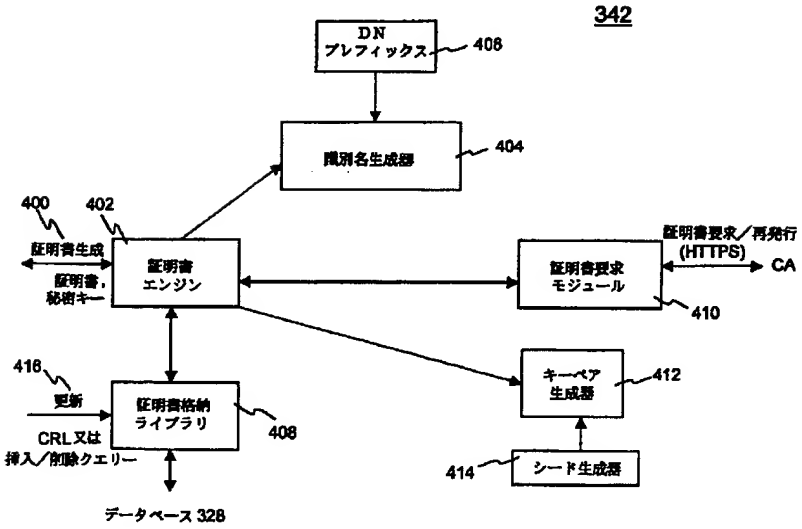
【図４】


352
プロクシサーバ
324
ユーザ アカウント (スミス, 123456) 861234567-10900
ユーザ アカウント (メアリ, 456789) 861234557-10910
302
モバイル デバイス 861234567 -10900
HDTP
(HDML)
HTTP
HTTP
356
通常の CA
358
特別の CA
354
ユーザ名とパスワード
HTTP
(HTML)
PC
314

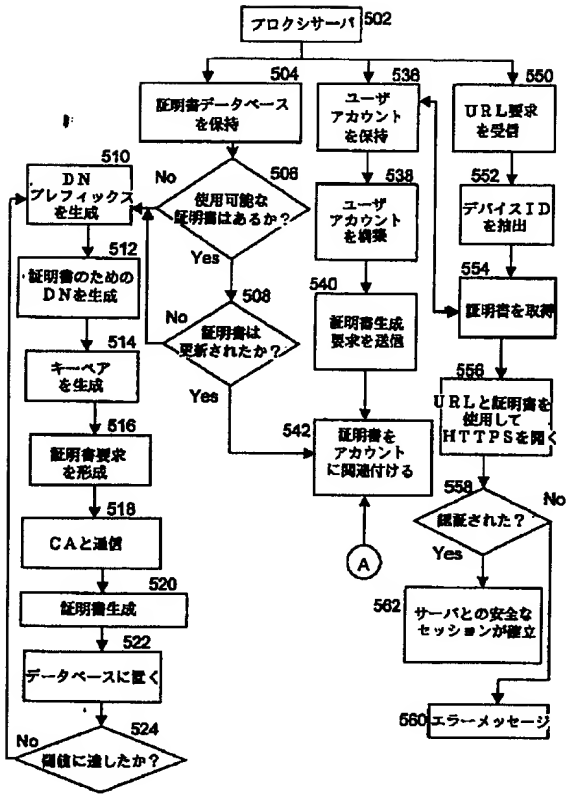
【図５】


320
デバイス ID
加入者 ID
ユーザ情報 (smith 123456)
証明書リスト
キー
661234567-10900_ pn.mobile.xyz.net
368
370
372
証明書
URL
374
CA S1による証明書
www.financial.com
376
CA S2による証明書
www.stock.com
378
CA S3による証明書
www.fund.com
380
CA S4による証明書
www.merchant.com
382
CA S5による証明書

【図６】


342
DN プレフィックス
406
識別名生成器
404
400
402
証明書生成
証明書, 秘密キー
証明書 エンジン
証明書要求 モジュール
証明書要求／再発行 (HTTPS)
CA
410
416
更新
CRL又は 挿入／削除クエリー
証明書格納 ライブラリ
408
キーペア 生成器
412
414
シード生成器
データベース 328

【図７】



フロントページの続き

(51) Int. Cl. [6]	識別記号	FI	
		H04L 9/00	675D